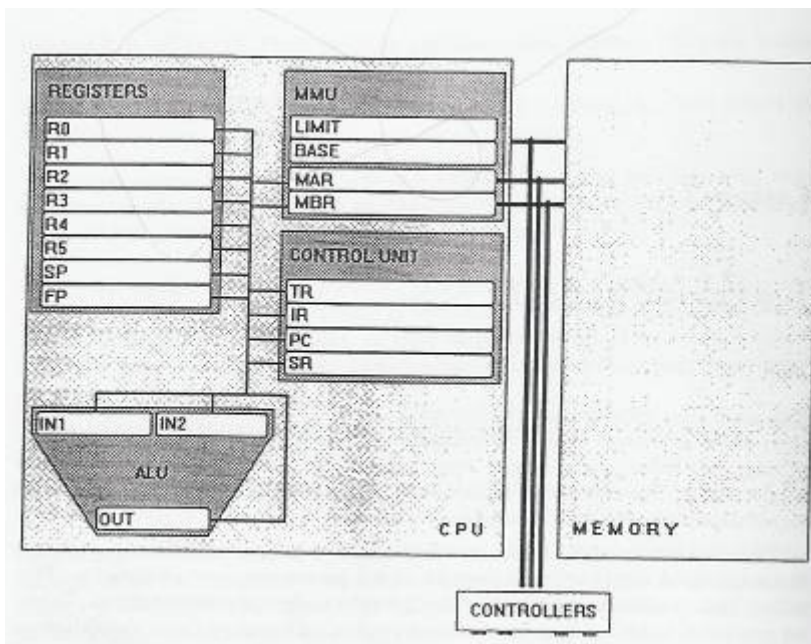


1. Keskusyksikön rakenne



Kuvassa on esitelty TTK-91 esimerkkikoneen keskusyksikkö. Oikeiden tietokoneiden keskusyksiköt ovat luonnollisesti monimutkaisempia, mutta tämä riittää oikein mainiosti asian havainnollistamiseen. Keskusyksikköön kuuluvat *suoritin* eli *prosessori* (central processing unit, CPU), *keskusmuisti* (memory, MEM) sekä *laiteohjaimet* (controller). Näitä osia liittävät toisiinsa *väylät* (bus).

1.1. Suoritin (CPU)

Tärkein keskusyksikön osista on ehdottomasti suoritin, joten käsittelemme sitä ensimmäiseksi. Suoritin noutaa konekäskyt muistista, tulkitsee ja lopuksi suorittaa ne, eli käsittelee dataa käskyjen määräämällä tavalla. Suoritin myös ohjaa kaikkea muuta toimintaa generoimiensa signaalien avulla.

Suorittimen tärkeimmät osat ovat *ohjausyksikkö* (control unit, CU), *aritmeettis-looginen yksikkö* (arithmetic-logical unit, ALU), ohjelmoijan käyttöön tarkoitetut *rekisterit* (registers) sekä *muistinhallintayksikkö* (memory management unit, MMU). Oikeissa tietokoneissa prosessoriin usein kuuluu liukulukuyksikkö liukulukuaritmetiikkaa varten, mutta TTK-91 käsittelee vain kokonaislukuja.

Ohjausyksikkö koostuu neljästä rekisteristä: TR eli *tilapäinen työrekisteri* (temporary register), IR eli *käskyrekisteri* (instruction register), PC eli *käskyosoitin* (program counter) ja SR eli *tilarekisteri* (state register).

Luentomonistetta ”Tietokoneen toiminta, Auvo Häkkinen, 1998” mukaillen.

- **Käskyosoittimessa** sijaitsee seuraavan konekäskyn osoite (joka on joko ”tämän käskyn osoite+1” tai hyppykäskyn osoite).
- Sillä hetkellä käsiteltänä oleva käsky sijaitsee **käskyrekisterissä**.
- **Tilapäisessä työrekisterissä** sijaitsevat ”väliaikaiset” tiedot, kuten käskyn tehollinen muistiosoite.
- **Tilarekisterissä** on paljon ”lippubittejä”, jotka kuvaavat virhetilanteita (esim. nollalla jakoa), tiloja (esim. etuoikeitettu tai keskeytykset estetty) ja vertailujen tuloksia (suurempi, pienempi, yhtäsuuri). Näiden lippujen nimet eivät ole tärkeitä (kunhan muistatte pari esimerkkiä siitä minkälaisia tarkoituksia niillä on), mutta ne ovat useimmiten intuitiivisia. Esim. suurempi kuin on G (Greater), nollalla jako on Z (divide by Zero) ja etuoikeitettu tila on P (Privileged state).

Aritmeettis-loogisessa yksikössä eli ALU:ssa suoritetaan kaikki aritmeettisia tai loogisia rakenteita käyttävät käskyt. Eli suomeksi kaikenlainen laskenta, vertailu ja bittijonoooperaatiot. ALU:ssa on kaksi sisäänmenoa operandeille ja yksi ulostulo tulokselle. Operaation tulos viedään ulostulosta edelleen johonkin rekisteriin.

Muistinhallintayksikkö eli MMU huolehtii kaikista viitteistä muistiin, sekä käskyistä laiteohjaimille. Se koostuu neljästä rekisteristä, *base*, *limit*, *MAR* ja *MBR*. Toteuttaakseen tehtävänsä sillä on liittymä suorittimen ulkopuoliseen väylään ja sitä kautta muistiin ja oheislaitteisiin.

- **Base** osoittaa prosessorin käyttöön annetun muistialueen alkuun. Kullakin prosessilla on oma muistialueensa, eikä se saa viitata sen ulkopuolelle. MMU muuntaa sille annetut suhteelliset osoitteet absoluuttisiksi osoitteiksi laskemalla ”Base + suhteellinen osoite”.
- **Limit** puolestaan on prosessin oman muistialueen yläraja. MMU tarkistaa ennen muistinoutoa, että annettu suhteellinen osoite ei ole suurempi kuin limit.
- **MAR** eli muistiosoiterekisteri (memory address register) on se paikka, johon laitetaan haettavan tiedon osoite, tai osoite johon tieto tallennetaan. MAR:ista osoite siirtyy osoite-väylälle.
- **MBR** eli muistin puskurirekisteri (memory buffer register) puolestaan on se paikka, johon talletettava tieto sijoitetaan, tai johon luettu tieto tulee muistista. MBR käyttää dataväylää tiedon siirtämiseen.

TTK-91:ssä ohjelmoijan käyttöön tarkoitettuja **rekistereitä** on kahdeksan. Kuusi *indeksirekisteriä* (R0-R5) sekä *pino-osoitin* (SP, stack pointer) ja *ympäristöosoitin* (FP, frame pointer).

- **Indeksirekisterit** ovat käytännöllisesti katsoen nopeita ”datapaikkoja” suorittimen sisällä. Niillä ei ole mitään erityistä tarkoitusta, joten ohjelmoija voi käyttää niitä miten haluaa.
- **Pino-osoitin** osoittaa pinon päällimmäistä alkiota. Teoriassa sitä voisi käsitellä ihan kuten indeksirekisteriäkin (lisätä, vähentää, kertoa, jne.), mutta koska sitä käytetään viitteenä pinon päällimmäiseen alkioon, ohjelman toiminta menee helposti sekaisin. PUSH käsky automaattisesti kasvattaa pino-osoittimen arvoa ja

Luentomonistetta ”Tietokoneen toiminta, Auvo Häkkinen, 1998” mukailten.

POP käsky puolestaan vähentää sitä. CALL ja SVC käskyt ja näistä palaavat EXIT ja IRET myös muuttavat pino-osoittimen arvoa.

- **Ympäristöosoitinta** käytetään lähinnä aliohjelmien kutsussa. Se osoittaa aktivointitietueen päälle, jolloin parametrit ja paluuarvon paikat voidaan löytää suhteessa siihen.

Oikeissa tietokoneissa on yleensä myös **välimuisti**. Välimuisti on suorittimella oleva muisti, joten se on huomattavasti nopeampi kuin muisti (keskumuisti), koska sen ei tarvitse käyttää suorittimen ulkopuolista väylää. Suorittimen sisäinen väylä on paljon nopeampi kuin suorittimen ulkoinen väylä. Välimuistissa pidetään viimeisimpien keskusmuistista noudettujen muistipaikkojen sisältöjä, ja usein myös niiden lähellä olleiden muistipaikkojen sisältöjä. Jos viitattavan muistipaikan sisältö on tässä välimuistissa, säästyy suoritin hitaammalta muistinoudolta.

1.2. Muisti

Muisti, eli keskusmuisti, on ohjelman suoritusaikana sekä käskyjen että datan talletuspaikka. Siellä sijaitsee myös käyttöjärjestelmä, sekä järjestelmään liittyvät kirjastorutiinit. Muistiin tieto on talletettu tavuina ja jokaisella tavulla (tai sanalla) on oma osoitteensa. Muistissa olevaan tietoon viitataan aina tällä muistiosoitteella.

1.3. Väylä

Väylä on siirtotie, johon voidaan liittää prosessori-, muisti- ja oheislaitteohjainkortteja. Kullakin väylään liitetyllä laitteella on oma osoitteensa, jonka se tunnistaa väylältä.

Väylän johtimet voidaan jakaa *data-*, *osoite-* ja *ohjausväylään* (data bus, address bus, control bus). Ohjausväylän avulla väylä varataan, ja sille annetaan ”lue” ja ”kirjoita” signaaleja. Osoiteväylälle tulee luonnollisesti osoite johon kirjoitetaan tai josta luetaan. Luettava tai kirjoitettava tieto kulkee dataväylää pitkin. Kaikki väylään kytketyt ohjaimet ”kuuntelevat” väylällä liikkuvia osoitteita. Kun esimerkiksi muistikortti havaitsee itseensä liittyvän osoitteen ja kirjoitusohjauksen, se siirtää tiedon väylältä omiin muistipiireihinsä.

1.4. Ohjaimet

Oheislaitteet kytketään prosessorin ja keskusmuistin yhteyteen *ohjaimien* välityksellä. Oheislaitteita tarvitaan tiedon syöttöä, tulostusta, pysyvää talletusta ja tiedonsiirtoa varten. Kutakin laitetyyppiä kohden on oma ohjaimensa ja yksi ohjain voi ohjata yhtä tai useampaa laitetta. Esimerkkejä ohjaimista ovat näytönohjain, levyohjain ja tietoliikenneohjain. Ohjaimia käytetään käyttöjärjestelmän palvelupyynnöillä.

2. Konekäskyjen suoritus

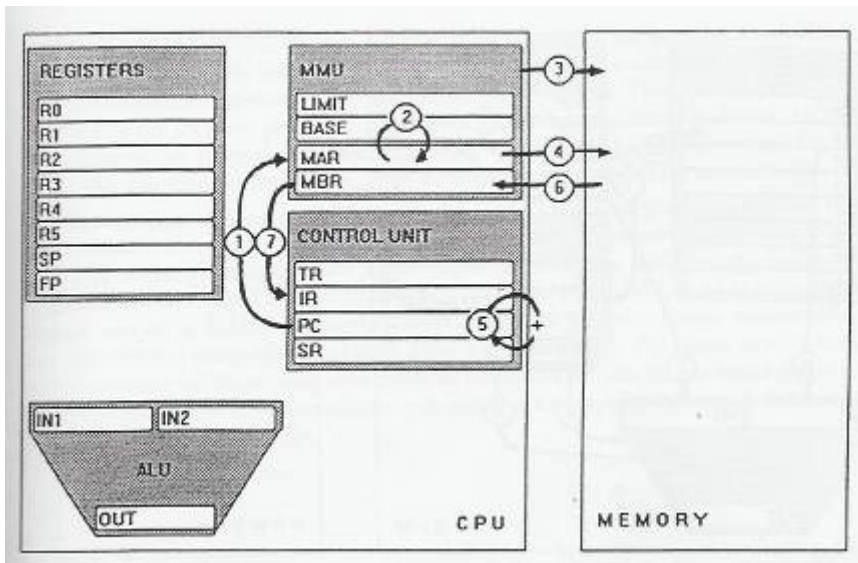
Proessori suorittaa tietojen käsittelyn sille annettujen ohjelmien mukaisesti. Nämä ohjelmat ovat kaikessa yksinkertaisuudessaan vain sarja yksittäisiä konekäskyjä. Koneen tunnistamat operaatiot on määritelty tietokoneen käskykannassa. Esimerkkejä ttk-91 koneen käskykannasta löytyy ”ttk-91 komentoesimerkkejä” dokumentista. Käskyjä laitteiston ohjaussignaaleiksi dekodattaessa on osattava erottaa operaatiokoodi, operandit sekä niiden tyyppi ja sijainti.

Yhden käskyn suoritus muodostuu useista vaiheista. Yhden käskyn aikana prosessori suorittaa seuraavia toimintoja:

1. Ohjausyksikkö CU siirtää käskyn osoitteen muistinhallintayksikön MMU muistiosoiterekisteriin ja valmistautuu uuden käskyn noutamiseen kasvattamalla käskyosoittimen (PC) arvoa.
2. MMU tarkistaa osoitteen oikeellisuuden ja tekee osoitemuunnoksen, sekä käynnistää muistinoudon.
3. MMU ottaa käskyn väylältä vastaan ja välittää sen edelleen ohjausyksikölle, joka tulkitsee sen.
4. Ohjausyksikkö selvittää käskyn operaatio-osan perusteella käskyssä tarvittavien operandien osoitteet ja noutaa datan muistista tai rekistereistä ALU:un.
5. Ohjausyksikkö suorittaa käskykoodin määräämät operaatiot tavallisimmin ALU:ssa ja tallettaa tulokset esimerkiksi rekistereihin.
6. Ohjausyksikkö tarkistaa tilarekisteristä SR oliko käskyn suorituksessa poikkeuksia tai aiheuttivatko prosessorin ulkopuoliset ohjaimet keskeytyksiä. Jos käskyn suoritusaikana tapahtui poikkeus tai keskeytys, siirtyy prosessori suorittamaan käyttöjärjestelmän koodia. **Huomioi, että yhden konekäskyn suoritusta ei voi keskeyttää ”kesken käskyn”!** Keskeytykset katsotaan vasta kun konekäsky on suoritettu.

Käsitellään seuraavaksi käskyn noutoa ja suoritusta vähän tarkemmin rekisteritasolla.

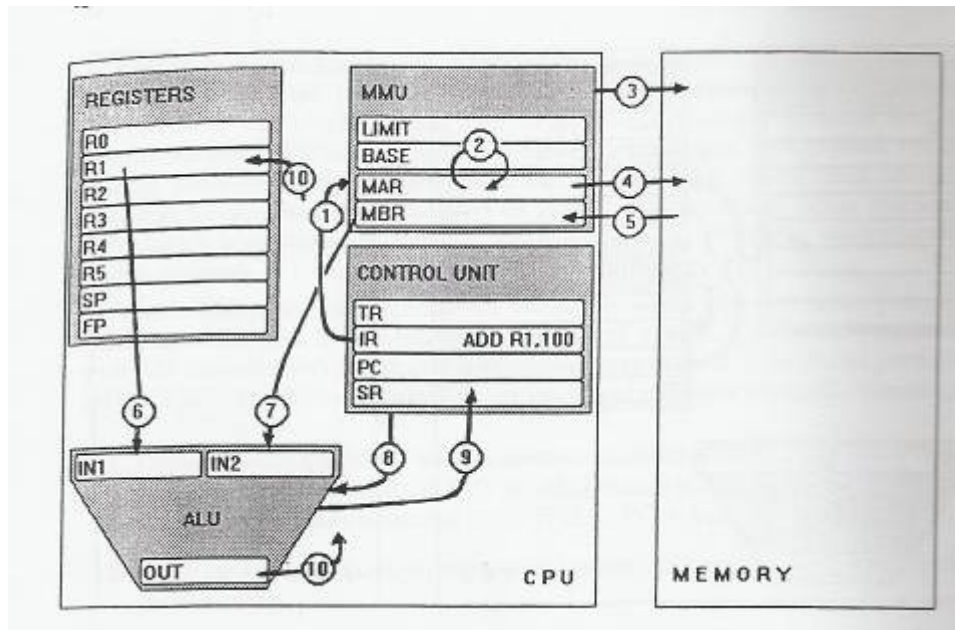
2.1. Käslyn nouto



Yllä olevassa kuvassa on esitetty käslynouto. Käslyn noutovaihe on kaikille käskyille samanlainen.

- 1) Ohjausyksikön lähettämät signaalit siirtävät käskyosoittimen PC arvon muistiosoiterekisteriin MAR.
- 2) MMU tekee osoitemuunnoksen (MAR:ista MAR:iin, absoluuttinen osoite = suhteellinen osoite + BASE).
- 3) Varataan väylä laittamalla käsky ”Lue” ohjausväylälle.
- 4) Siirretään muistiosoiterekisterissä oleva osoite osoiteväylälle.
- 5) Kasvatetaan PC:n arvoa yhdellä. Tämä voisi tapahtua missä tahansa vaiheessa sen jälkeen kuin vanha PC on siirretty MAR:iin mutta ennen kuin käskyä ruvetaan suorittamaan.
- 6) Hetken kuluttua osoitetun muistipaikan sisältö siirtyy dataväylää pitkin muistin puskurirekisteriin MBR.
- 7) MBR:stä käsky siirretään edelleen käskyrekisteriin IR. Ja käslyn käsittely voi alkaa.

2.2. Käsken suoritus



Yllä olevassa kuvassa on esitetty esimerkikäsken "ADD R1, 100" suoritus. Käsken suorituksen kulku riippuu siitä millaisesta käskystä on kyse, mutta se on suunnilleen samantapainen kaikille käskyille.

Käskynoudon jälkeen käsky on rekisterissä IR. Käskykoodin perusteella ohjausyksikkö löytää mikrokoodista käsken suorittamisessa tarvittavat ohjaustiedot (jokaista bittiähan vastasi yksi piuha, ja ne on laitettu kimppuihin siten, että operaatiokoodi ja operandit tulevat erikseen). Tässä tapauksessa käsky on yhteenlaskukäsky, ja sen operandit ovat rekisterissä R1 ja muistiosoitteessa 100. Tulos on talletettava rekisteriin R1.

- 1) Koska jälkimmäinen operandi on muistissa, on tehtävä muistinouto. Ohjausyksikön lähettämien signaalien ansiosta osoite "100" siirtyy käskyrekisteristä muistiosoiterekisteriin MAR.
- 2) MMU tarkistaa osoitteen oikeellisuuden ja tekee osoitemuunnoksen osoite = suhteellinen osoite + BASE.
- 3) Ohjauskäsky "Lue" käynnistää muistinoudon...
- 4) ... ja absoluuttinen (eli fyysinen) osoite siirretään osoiteväylälle.
- 5) Hetken kuluttua osoitetun muistipaikan sisältö on muistipuskurirekisterissä MBR.
- 6) Nyt kun molemmat operandit ovat prosessorin sisällä, käsky voidaan suorittaa. Rekisterin R1 arvo siirretään ALU:n sisäänmenoon IN1.
- 7) Muistipaikan 100 sisältö siirretään MBR:stä ALU:n toiseen sisäänmenoon IN2.
- 8) Ohjausyksikkö antaa käsken "yhteenlaske".
- 9) Jos käsken laskutoimituksessa sattui yli- tai alivuoto, merkitään tieto siitä tilarekisteriin SR.
- 10) Lopuksi yhteenlaskun tulos tulee ulos ALU:n ulostulosta ja siirtyy rekisteriin R1.